

Appl. No. 09/965,002
Response to Notice of Non-Compliant Appeal Brief dated 01/10/2008
Reply to Office Action of 12/10/2007

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re: Application of:	:
Abdelhadi et al.	:
	: Before the Examiner:
Serial No: 09/965,002	: Michael D. Meucci
	:
Filed: 09/27/2001	: Group Art Unit: 2142
	:
Title: APPARATUS AND METHOD	: Confirmation No.: 2728
OF REPRESENTING REAL-TIME	:
DISTRIBUTED COMMAND	:
EXECUTION STATUS ACROSS	:
DISTRIBUTED SYSTEMS	:

RESPONSE TO NOTICE OF NON-COMPLIANT APPEAL BRIEF

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

This is a Response to a Notice of Non-Compliant Appeal Brief dated
December 10, 2007.

AUS920010905US1

BRIEF FOR APPLICANTS - APPELLANTS

(i)

Real Party in Interest

The real party in interest is International Business Machines Corporation (IBM), the assignee.

(ii)

Related Appeals and Interferences

There are no other appeals or interferences known to appellants, appellants' representative or assignee, which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(iii)

Status of Claims

Claims 1 - 37 have been finally rejected. Claims 1 – 37 are being appealed.

(iv)

Status of Amendment

An "Amendment-After-Final" was not filed.

(v)

Summary of Claimed Subject Matter

The invention, as claimed in Claim 1, provides a method of displaying an execution status of a command that is sent to a plurality of computer systems on a network for execution. The method comprises the steps of: displaying a dialog window (page 17, lines 9 - 16 and execution progress window 1000 of Fig. 10), said dialog window being divided into sub-windows for displaying present status of the execution of the command on each of the computer systems (page 17, line
AUS920010905US1

24 to page 18, line 19 and "waiting" sub-window 1005, "working" sub-window 1010, "successful" sub-window 1015 and "failed" sub-window 1020 of Fig. 10); and displaying the status of the execution of the command on each of the computer systems within a proper sub-window (page 17, line 28 to page 18, line 19).

The invention, as claimed in Claim 13, provides a computer program product on a computer readable medium for displaying an execution status of a command sent to a plurality of computer systems on a network for execution. The computer program product comprises: code for displaying a dialog window (page 17, lines 9 - 16 and execution progress window 1000 of Fig. 10), said dialog window being divided into sub-windows for displaying present status of the execution of the command on each of the computer systems (page 17, line 24 to page 18, line 19 and "waiting" sub-window 1005, "working" sub-window 1010, "successful" sub-window 1015 and "failed" sub-window 1020 of Fig. 10); and code for displaying the status of the execution of the command on each of the computer systems within the proper sub-window (page 17, line 28 to page 18, line 19). The code means of the claim are the steps outlined on page 23, line 15 to page 25, line 8 as well as Fig. 12.

The invention, as claimed in Claim 25, provides an apparatus for displaying an execution status of a command sent to a plurality of computer systems on a network for execution. The apparatus comprises: means for displaying a dialog window (page 17, lines 9 - 16 and execution progress window 1000 of Fig. 10), said dialog window being divided into sub-windows for displaying present status of the execution of the command on each of the computer systems (page 17, line 24 to page 18, line 19 and "waiting" sub-window 1005, "working" sub-window 1010, "successful" sub-window 1015 and "failed" sub-window 1020 of Fig. 10); and means for displaying the status of the execution of the command on each of the computer systems within the proper sub-window (page 17, line 28 to page 18, line 19). The means of the claim are

the steps outlined on page 23, line 15 to page 25, line 8 as well as Fig. 12 when executed by processor 202, 204 of Fig. 2 or 302 of Fig. 3.

The invention, as claimed in Claim 37, provides a method of displaying an execution status of a command that is being executed by a plurality of computer systems on a network on which different system management software utilities having different command structures are running. The method comprises the steps of: enabling a user to enter the command in a common interface, the command being either a request to start execution of another command or to stop execution of the other command, the common interface translating the command into the different command structures (page 13, lines 16 – 21 and box 606 of Fig. 6); enabling a user to send the command to the plurality of the computer systems (page 14, lines 1 – 28 and slider 710 and box 720 of Fig. 7); enabling a user to indicate whether or not the execution of the command is to be monitored (page 15, lines 4 – 8 and box 740 of Fig. 4) ; displaying, if the execution of the command is to be monitored, a dialog window (page 17, lines 9 - 16 and execution progress window 1000 of Fig. 10) that is divided into a waiting, working, successful and failed sub-windows for displaying present status of the execution of the command on each of the computer systems executing the command (page 17, line 24 to page 18, line 19 and “waiting” sub-window 1005, “working” sub-window 1010, “successful” sub-window 1015 and “failed” sub-window 1020 of Fig. 10); and displaying the status of the execution of the command on each of the computer systems within a proper sub-window (page 17, line 28 to page 18, line 19).

(vi)

Grounds of Rejection to be Reviewed on Appeal

Whether Claims 1 – 8, 13 – 20, 25 - 32 and 37 were properly rejected under 35 U.S.C. §103(a) as being unpatentable over Joyce et al. in view of Ahmed et al.

Whether Claims 9, 21 and 33 were properly rejected under 35 U.S.C. §103(a) as being unpatentable over Joyce et al. in view of Ahmed et al. and further in view of Kimura et al.

Whether Claims 10 – 12, 22 – 24 and 34 - 36 were properly rejected under 35 U.S.C. §103(a) as being unpatentable over Joyce et al. in view of Ahmed et al. and Kimura et al. and further in view of Darland et al.

(vii)

Arguments

Whether Claims 1 – 8, 13 – 20, 25 - 32 and 37 were properly rejected under 35 U.S.C. §103(a) as being unpatentable over Joyce et al. in view of Ahmed et al.

In considering a Section §103 rejection, the subject matter of the claim “as a whole” must be considered and analyzed. In the analysis, it is necessary that the scope and contents of the prior art and differences between the art and the claimed invention be determined. *Graham v. John Deere Co.*, 383 U.S. 1 (1966).

Claims 1, 13, 25 and 37

In the Examiner’s Answer Brief of August 03, 2006, the Examiner admitted that Joyce et al. do not teach the window being divided into sub-windows for displaying present status of the execution of the command on each of the computer systems. However, the Examiner asserted that Ahmed et al. (US Patent No. 6,647,432) discloses: “Windowing software technology is applied where it is important for an operator to display and interact with multiple programs executing concurrently in a computer system comprising one or more interconnected workstations.” Therefore, the Examiner concluded it would have been obvious for one skilled in the art to combine the teachings of Joyce et al. with those of Ahmed et al. to arrive at the claimed invention.

Firstly, whether or not Ahmed et al. disclose a "Windowing software technology that is applied where it is important for an operator to display and interact with multiple programs executing concurrently in a computer system comprising one or more interconnected workstations" is irrelevant as the claim limitations specifically include displaying a dialog window, said dialog window being divided into sub-windows for displaying present status of the execution of the command on each of the computer systems; and displaying the status of the execution of the command on each of the computer systems within a proper sub-window.

Secondly, Joyce et al. purport to teach an extensible, distributed monitoring system for monitoring a distributed application system (i.e., a collection of processes concurrently executing on different computer systems (or nodes) that work together to accomplish a task. In accordance with the teachings of Joyce et al., the extensible, distributed monitoring system uses a multilingual inter-process communication (IPC) facility, a window system, a hierarchical graphics package, an interactive graphics editor and a distributed monitoring system.

The IPC facility is used to facilitate exchanges of messages among the processes (see Fig. 3). The graphics package provides routines for creating and manipulating pictures and the graphics editor facilitates the creation of pictures that can be used to represent specific states of an executing distributed program. The window system enables a system of processes spanning multiple machines to be observed and controlled from a workstation (see Section 2.1).

The exchanges of messages among the processes may be monitored through traces that can be either textual or graphical. A trace is a depiction of communication events occurring in the distributed system. A textual trace reports each event in an event stream by including the name of a process that initiates the event, the event type, the name of the process that is the subject of the event and if the event is a message, the content of the message (see Section 3.1). A graphical trace is an animated graphical view of an event stream.

AUS920010905US1

Whenever an event is received, a picture that represents the current state of an inter-process communication is updated (see Section 3.2). This results in providing to a user an animated graphical view of an event stream, such as that shown in Fig. 7.

Thus, Joyce et al. disclose a method of displaying the status of messages between processes in a message stream.

By contrast, the present invention teaches a method of displaying the status of one command that is being executed simultaneously on each one of a plurality of computer systems.

Consequently, contrary to the Examiner's assertion, Joyce et al. do not teach the step of ***displaying present status of the execution of the command on each of the computer systems*** as in the claimed invention.

Thirdly, Ahmed et al. purport to teach a distributed framework for inter-task communication between workstation applications. According to the purported teachings of Ahmed et al., one or more workstations are interconnected by an extensible inter-task communication (ITC) apparatus. Each workstation has a display in which one or more windows are presented to an operator. Each window is generated in response to the execution of an application program or client application. Each client application has a Human Interface Code and a Framework Code. The Framework Code, in conjunction with a server program, transmits and communicates event information directly between a first client application and a second client application, or a plurality of client application programs concurrently executing in one or more workstations of a network of interconnected workstations, without requiring that event information pass through and register with an intervening server or dispatcher application program, if and when an interest object is initially transmitted between the first client application and the second client application via the server program.

An event is an action taken by one operator at a workstation. For example, that operator may drag the cursor by moving a mouse or perhaps the operator may delete data or create new data. That event information, being
AUS920010905US1

practiced by one operator in one program application at one workstation, may be needed by another operator in another program application at another workstation. The inter-process communication can transmit that event information from the one program application to all other program applications in the network of workstations, without requiring that the event information register with an intervening server or dispatcher program, provided that an interest object(s) was initially transmitted between the one program application and all the other program applications via a server which are concurrently executing in all of the workstations in the network of workstations.

In the BACKGROUND OF INVENTION Section (particularly in col. 2, line 66 to col. 3, line 64), Ahmed et al. delineate reasons why a method of exchanging messages using a facility such as that disclosed in Joyce et al. was not ideal and proceed to show why their method is superior (see col. 4, lines 1 – 12). Thus, Ahmed et al. acknowledge the existence of the method in Joyce et al. and specifically teach away from it.

A prior art reference may be considered to teach away when "a person of ordinary skill, upon reading the reference, would be discouraged from following the path set out in the reference, or would be led in a direction divergent from the path that was taken by the applicant." *In re Gurley*, 27 F.3d 551, 553, 31 USPQ 2d 1130, 1131 (Fed. Cir. 1994). See also *Monarch Knitting Mach. Corp. v. Sulzer Morat GmbH*, 45 USPQ 2d 1977, 1984 (Fed. Cir. 1998).

In this case, since Ahmed et al. acknowledge the existence of the Joyce et al. method (i.e., exchanging messages through the IPC facility) and specifically teach that it is not to be used for lack of efficiency and control.

The Courts have repeatedly held that references that teach away cannot serve to create a prima facie case of obviousness. *In re Gurley*, 27 F.3d 551, 553, 31 USPQ 2d 1130 (Fed. Cir. 1994), in *McGinley v. Franklin Sports Inc.*, 60 USPQ 2d 1001, 1010 (Fed. Cir. 2001). See also *KSR International Co. v. Teleflex Inc.*, 550 U.S. ____, 82USPQ2d 1385 (2007).

Hence, Appellants submit that the Examiner has impermissibly combined the teachings of Joyce et al. with those of Ahmed et al. in a quest to render the claimed invention obvious.

Nonetheless, even if the teachings of Joyce et al. could be combined with those of Ahmed et al., the resulting combination would not teach the claimed invention.

Ahmed et al. advocate displaying each executing application program in a window on a screen of a computer system. But note that the graphic images in the disclosure of Joyce et al. are a succession of images that are displayed to a user to provide the animated graphical view of an event stream when processes are exchanging messages (see Fig. 7 on page 134). Consequently, the images are not displayed at the same time to the user.

Therefore, combining the teachings of Joyce et al. with those of Ahmed et al. would show a screen on which a plurality of windows are used to display a plurality of executing application programs and on which a graphical trace may be displayed through a succession of images to provide an animated graphical view of a communication event between the application programs.

However, the combination would not teach the steps of (1) displaying a dialog window, said dialog window being divided into sub-windows for displaying present status of the execution of the command on each of the computer systems; and (2) displaying the status of the execution of the command on each of the computer systems within a proper sub-window as asserted by the Examiner.

Based on the above arguments, Appellants submit that the claims are not obvious in view of the teachings of the applied references.

Claims 2, 14 and 26

Claims 2, 14 and 26 include the limitations "wherein said sub-windows include a "waiting" sub-window, a "working" sub-window and a "completed" sub-window."

AUS920010905US1

The Examiner asserted that on page 133, line 41 to page 134, line 6, Joyce et al. teach that when the status of a machine changes, the display is changed to reflect the new status. Therefore, the Examiner concluded, although Joyce et al. do not explicitly teach displaying the names of the computer systems in the sub-windows in accordance with the status of the execution of the command on the computer systems, it would have been obvious to one of ordinary skill in the art to modify Joyce et al.'s method by placing the machine name icons into separate window based on their current state of waiting, receiving or finished.

However, it should be noted that the display is changed to reflect a new status of a message transfer between two processes which results in a succession of images being displayed. This is what allows for the animated graphical view of the event stream. Consequently, no more than one window is used to display the statuses of message transfers among the processes.

Hence, even if the Examiner was correct in asserting that the teachings of Joyce et al. could be modified to show the machine name icons into separate window based on their current state of waiting, receiving or finished, the resulting modification of the teachings of Joyce et al. would not teach the step of **displaying a dialog window being divided into sub-windows wherein said sub-windows include a "waiting" sub-window, a "working" sub-window and a "completed" sub-window for displaying present status of the execution of the command on each of the computer systems** as in the claimed invention.

Therefore, Appellants submit that the claims are not obvious by the teachings of the applied references.

Claims 4, 16 and 28

Claims 4, 16 and 28 include the limitations "wherein when the command begins to execute on a computer system, the name of the computer system is moved from the "waiting" sub-window to the "working" sub-window."

AUS920010905US1

In rejecting the instant claims, the Examiner used the same rationale used to reject Claims 2, 14 and 26. Appellants respectfully disagree.

Neither Joyce et al. nor Ahmed et al. teach, show or so much as suggest the use of “**waiting**” sub-window and “**working**” sub-window much less the step of moving the name of a computer system from the “waiting” sub-window to the “working” sub-window when the command begins to execute on the computer system.

Thus, Appellants submit that Claims 4, 16 and 28 are not obvious by the teachings of the applied references.

Claims 5, 17 and 29

Claims 5, 17 and 29 include the limitations “wherein when the command has finished executing on a computer, the name of the computer is moved from the “working” sub-window to the “completed” sub-window.”

Again, in rejecting the instant claims, the Examiner used the same rationale used to reject Claims 2, 14 and 26. Again, Appellants respectfully disagree.

Neither Joyce et al. nor Ahmed et al. teach, show or so much as suggest the use of “**working**” sub-window and “**completed**” sub-window much less the step of moving the name of a computer system from the “working” sub-window to the “completed” sub-window when the command has finished to execute on the computer system.

Thus, Appellants submit that Claims 5, 17 and 29 are not obvious by the teachings of the applied references and request reversal of the rejection.

Claims 6, 18 and 30

Claims 6, 18 and 30 include the limitations “wherein the “completed” sub-window is further divided into a “successful” sub-window and a “failed” sub-window.”

Once more, the Examiner used the same rational used to reject Claims 2, 14 and 26 to reject the present claims. Once more, Appellants respectfully disagree.

Neither Joyce et al. nor Ahmed et al. teach, show or so much as suggest the use of a **“completed”** window being sub-divided into a **“successful”** sub-window and **“failed”** sub-window.

Thus, Appellants submit that Claims 6, 18 and 30 are not obvious by the teachings of the applied references and request reversal of the rejection.

Claims 7, 19 and 31

Claims 7, 19 and 31 include the limitations “wherein the names of the computer systems that have successfully completed the execution of the command are displayed in the “successful” sub-window.”

The Examiner rejected the instant claims under the same rational as the rejection of Claims 2, 14 and 26. Appellants continue to respectfully disagree.

As discussed above, neither Joyce et al. nor Ahmed et al. teach, show or so much as suggest the use of a **“completed”** window being sub-divided into a **“successful”** sub-window and **“failed”** sub-window. Consequently, the combination of their teachings cannot include the teachings of **displaying the names of the computer systems that have successfully completed the execution of the command in the “successful” sub-window** as claimed.

Thus, Appellants submit that Claims 7, 19 and 31 are not obvious by the teachings of the applied references and request reversal of the rejection.

Claims 8, 20 and 32

Claims 8, 20 and 32 include the limitations “wherein the names of the computer systems that have not successfully completed the execution of the command are displayed in the “failed” sub-window.”

The Examiner rejected the instant claims under the same rational as the rejection of Claims 2, 14 and 26. Appellants respectfully disagree.

AUS920010905US1

As mentioned above, neither Joyce et al. nor Ahmed et al. teach, show or so much as suggest the use of a **“completed”** window being sub-divided into a **“successful”** sub-window and **“failed”** sub-window. Consequently, the combination of their teachings cannot include the teachings of **displaying the names of the computer systems that have not successfully completed the execution of the command in the “failed” sub-window** as claimed.

Thus, Appellants submit that Claims 8, 20 and 32 are not obvious by the teachings of the applied references and request reversal of the rejection.

Whether Claims 9, 21 and 33 were properly rejected under 35 U.S.C. §103(a) as being unpatentable over Joyce et al. in view of Ahmed et al. and further in view of Kimura et al.

Claims 9, 21 and 33 include the limitations “wherein the names of the computer systems that have not successfully completed the execution of the command are displayed in red in the “failed” sub-window.”

The Examiner admitted that neither Joyce et al. nor Ahmed et al. disclose that the names of the computer systems are displayed in red in the “failed” sub-window. However, the Examiner asserted that in col. 9, lines 56 – 60 Kimura et al. teach that a color such as red can be used to denote an error condition in a display.

Firstly, whether or not “Kimura et al. teach that a color such as red can be used to denote an error condition in a display” is irrelevant. The claims specifically include the limitations “the names of the computer systems that have not successfully completed the execution of the command are displayed in red in the “failed” sub-window.” They do not include the limitations of using a color such as red to denote an error condition.

Secondly, Appellants would like to point out that in col. 9, lines 56 – 60 Kimura et al. disclose that when an error occurs in a video/audio device shown on a display, the **background color** is changed from white to a warning color, such as red flickers.

AUS920010905US1

Kimura et al. do not teach displaying **the names of the computer systems that have not successfully completed the execution of the command in red in the “failed” sub-window.**

Thirdly, Kimura et al. purport to teach an error monitoring system for video/audio devices. According to the teachings of Kimura et al., the system is comprised of a processing unit for executing an error monitoring process by detecting errors occurring in the video/audio devices, a communication unit connecting the video/audio devices to the processing unit, and a display unit connected to the processing unit to simultaneously display on a common display plane a plurality of images indicating the video/audio devices and to give an error indication in accordance with a result of the error monitoring process by the processing unit.

Thus, there is no reason for anyone to combine the teachings of Kimura et al. with those of Joyce et al. and Ahmed et al. absent some teaching or suggestion to do so.

In *In re Fritch*, 972 F.2d 1260, 23 USPQ 2d 1780, 1783–84 (Fed. Cir. 1992), the Court ruled that “[o]bviousness cannot be established by combining the teachings of the prior art to produce the claimed invention, absent some teaching or suggestion supporting the combination. Under section 103, teachings of references can be combined *only* if there is some suggestion or incentive to do so.” (quoting *ACS Hosp. Systems, Inc. v. Montefiore Hosp.*, 732 F.2d 1572, 1577, 221 USPQ 929, 933 (Fed. Cir. 1984)). . . . The mere fact that the prior art may be modified in the manner suggested by the Examiner does not make the modification obvious unless the prior art suggested the desirability of the modification.

As mentioned above, the teachings of Joyce et al. are directed toward an extensible, distributed monitoring system for monitoring a distributed application system while those of Ahmed et al. are directed toward a distributed framework for inter-task communication between workstation applications. By contrast, the

teachings of Kimura et al. are directed toward an error monitoring system for video/audio devices.

There is not any language in the disclosure of Joyce et al., the one of Ahmed et al. or of Kimura et al. suggesting or teacheang combining the teachings of the references together.

Consequently, Appellants submit that the Examiner has impermissibly combined the teachings of Joyce et al. and Ahmed et al. with those of Kimura et al. in a quest to arrive at the claimed invention. Hence the claims are not patentable over the applied references.

Hence, Appellants submit that the claims are not obvious by the teachings of the applied references.

Whether Claims 10 – 12, 22 – 24 and 34 - 36 were properly rejected under 35 U.S.C. §103(a) as being unpatentable over Joyce et al. in view of Ahmed et al. and Kimura et al. and further in view of Darland et al.

Claims 10, 22 and 34

Claims 10, 22 and 34 include the limitations “wherein when the displayed name of a computer system is selected further information about the status of the command executing on the computer system is displayed.”

The Examiner, as in the case of the previous dependent claims, admitted that the previously applied references do not teach the step of displaying further information about the status of the command executing on the computer system when the displayed of a computer system is selected. However, the Examiner asserted that Darland et al. teach in col. 11, lines 11, 12 and 18 – 22 that additional operating information about an item can be obtained by selecting that item.

Firstly, whether additional operating information about an item can be obtained by selecting that item is irrelevant since the claimed limitations specifically state that when the displayed name of a computer system is selected

AUS920010905US1

further information about the status of the command executing on the computer system is displayed.

Secondly, Darland et al. disclose in col. 11, lines 6 – 37:

The following procedure is used to access the Single Operator display:

1. The user points to the terminal icon for the operator he wishes to display.
2. The user holds the SHIFT key of the

If computer and clicks the mouse button on the terminal icon. this is done for a terminal which is shown as OFFLINE or for an operator who is not presently signed on to a job, the following message is provided:

The monitoring system displays the Single Operator display screen, as shown in FIG. 9. When the display appears, the various data fields begin to display information about the operator extracted from the Voicelink system. Next, the operator's bar chart fills in to show the operator's calling results across six selected release codes. Finally, the operator's productivity chart updates to show the use of the operator's time on the system.

It is noted that the Overview screen can show an operator active (indicated by the letter "A" in the terminal icon) on a job who has quit that job since the last update. When this happens, the computer monitor provides the message:

The user can click the mouse button on the "OK" option to return to the Overview display and select another operator.

Thus, in the passages cited by the Examiner, Darland et al. disclose a method of accessing a display. However, Darland et al. do not teach, show or so much as suggest the step of displaying **further information about the status of the command executing on the computer system when the name of a computer system is selected.**

Consequently, combining the teachings of Joyce et al., Ahmed et al. and Kimura et al. with those of Darland et al. does not teach the claimed invention.

Claims 11, 23 and 35

Claims 11, 23 and 35 include the limitations "wherein if the selected computer system is displayed in the failed sub-window, a reason for the unsuccessful completion of the execution of the command is displayed."

In this case, the Examiner asserted that Kimura et al. further disclose in col. 10, lines 9 – 18 that when an error condition occurs, an error code and an error message can be displayed.

Appellants would like to point out that the claimed invention specifically call for displaying a reason for the unsuccessful completion of the execution of the command if the selected computer system is displayed in the failed sub-window.

Therefore, even if the Examiner were correct in asserting that Kimura et al. disclose that when an error condition occurs, an error code and an error message can be displayed, the combination of the teachings of Joyce et al., Ahmed et al., Kimura et al. and Darland et al. would still not teach the claimed invention.

Hence, Appellants submit that the claims are not obvious by the teachings of the applied references.

Claims 12, 24 and 36

Claims 12, 24 and 36 include the limitations "wherein if the selected computer system is displayed in the executing sub-window, a real-time progress of the execution of the command is displayed."

The Examiner asserted that Darland et al. further disclose that the additional operating information obtained by selecting the item can include a real-time progress indicator in col. 11, lines 2 and 24 – 26.

Again, Appellants would like to point out that the claimed invention specifically state that a real-time progress of the execution of the command is displayed if the selected computer system is displayed in the executing sub-window.

Therefore, even if the Examiner were correct in asserting that Darland et al. further disclose that the additional operating information obtained by selecting the item can include a real-time progress indicator, the combination of the teachings of Joyce et al., Ahmed et al., Kimura et al. and Darland et al. would still not teach the claimed invention.

Hence, Appellants submit that the claims are not obvious by the teachings of the applied references.

Based on the above-proffered arguments, Appellants respectfully request reversal of the rejection of the claims in the Application.

Respectfully Submitted

By: 

Volel Emile
Attorney for Applicants
Registration No. 39,969
(512) 306-7969

(viii)
Claims Appendix

1. (Previously presented) A method of displaying an execution status of a command, said command being sent to a plurality of computer systems on a network for execution, said method comprising the steps of:

displaying a dialog window, said dialog window being divided into sub-windows for displaying present status of the execution of the command on each of the computer systems; and

displaying the status of the execution of the command on each of the computer systems within a proper sub-window.

2. (Original) The method of Claim 1 wherein said sub-windows include a "waiting" sub-window, a "working" sub-window and a "completed" sub-window.
3. (Original) The method of Claim 2 wherein the step of displaying the status of the execution of the command includes displaying the names of the computer systems in the sub-windows in accordance with the status of the execution of the command on the computer systems.
4. (Original) The method of Claim 3 wherein when the command begins to execute on a computer system, the name of the computer system is moved from the "waiting" sub-window to the "working" sub-window.
5. (Original) The method of Claim 4 wherein when the command has finished executing on a computer, the name of the computer is moved from the "working" sub-window to the "completed" sub-window.

6. (Original) The method of Claim 5 wherein the "completed" sub-window is further divided into a "successful" sub-window and a "failed" sub-window.
7. (Original) The method of Claim 6 wherein the names of the computer systems that have successfully completed the execution of the command are displayed in the "successful" sub-window.
8. (Previously presented) The method of Claim 7 wherein the names of the computer systems that have not successfully completed the execution of the command are displayed in the "failed" sub-window.
9. (Previously presented) The method of Claim 8 wherein the names of the computer systems that have not successfully completed the execution of the command are displayed in red in the "failed" sub-window.
10. (Original) The method of Claim 9 wherein when the displayed name of a computer system is selected further information about the status of the command executing on the computer system is displayed.
11. (Original) The method of Claim 10 wherein if the selected computer system is displayed in the failed sub-window, a reason for the unsuccessful completion of the execution of the command is displayed.
12. (Previously presented) The method of Claim 11 wherein if the selected computer system is displayed in the executing sub-window, a real-time progress of the execution of the command is displayed.
13. (Previously presented) A computer program product on a computer readable medium for displaying an execution status of a command, said

AUS920010905US1

command being sent to a plurality of computer systems on a network for execution, said computer program product comprising:

code for displaying a dialog window, said dialog window being divided into sub-windows for displaying present status of the execution of the command on each of the computer systems; and

code for displaying the status of the execution of the command on each of the computer systems within the proper sub-window.

14. (Original) The computer program product of Claim 13 wherein said sub-windows include a "waiting" sub-window, a "working" sub-window and a "completed" sub-window.
15. (Original) The computer program product of Claim 14 wherein the code for displaying the status of the execution of the command includes code for displaying the names of the computer systems in the sub-windows in accordance with the status of the execution of the command on the computer systems.
16. (Original) The computer program product of Claim 15 wherein when the command begins to execute on a computer system, the name of the computer system is moved from the "waiting" sub-window to the "working" sub-window.
17. (Original) The computer program product of Claim 16 wherein when the command has finished executing on a computer, the name of the computer is moved from the "working" sub-window to the "completed" sub-window.

18. (Original) The computer program product of Claim 17 wherein the "completed" sub-window is further divided into a "successful" sub-window and a "failed" sub-window.
19. (Original) The computer program product of Claim 18 wherein the names of the computer systems that have successfully completed the execution of the command are displayed in the "successful" sub-window.
20. (Previously presented) The computer program product of Claim 19 wherein the names of the computer systems that have not successfully completed the execution of the command are displayed in the "failed" sub-window.
21. (Previously presented) The computer program product of Claim 20 wherein the names of the computer systems that have not successfully completed the execution of the command are displayed in red in the "failed" sub-window.
22. (Original) The computer program product of Claim 21 wherein when the displayed name of a computer system is selected further information about the status of the command executing on the computer system is displayed.
23. (Original) The computer program product of Claim 22 wherein if the selected computer system is displayed in the failed sub-window, a reason for the unsuccessful completion of the execution of the command is displayed.
24. (Previously presented) The computer program product of Claim 23 wherein if the selected computer system is displayed in the executing sub-

window, a real-time progress of the execution of the command is displayed.

25. (Previously presented) An apparatus for displaying an execution status of a command, said command being sent to a plurality of computer systems on a network for execution, said apparatus comprising:

means for displaying a dialog window, said dialog window being divided into sub-windows for displaying present status of the execution of the command on each of the computer systems; and

means for displaying the status of the execution of the command on each of the computer systems within the proper sub-window.

26. (Original) The apparatus of Claim 25 wherein said sub-windows include a "waiting" sub-window, a "working" sub-window and a "completed" sub-window.
27. (Original) The apparatus of Claim 26 wherein the means for displaying the status of the execution of the command includes means for displaying the names of the computer systems in the sub-windows in accordance with the status of the execution of the command on the computer systems.
28. (Original) The apparatus of Claim 27 wherein when the command begins to execute on a computer system, the name of the computer system is moved from the "waiting" sub-window to the "working" sub-window.
29. (Original) The apparatus of Claim 28 wherein when the command has finished executing on a computer, the name of the computer is moved from the "working" sub-window to the "completed" sub-window.

30. (Original) The apparatus of Claim 29 wherein the "completed" sub-window is further divided into a "successful" sub-window and a "failed" sub-window.
31. (Original) The apparatus of Claim 30 wherein the names of the computer systems that have successfully completed the execution of the command are displayed in the "successful" sub-window.
32. (Previously presented) The apparatus of Claim 31 wherein the names of the computer systems that have not successfully completed the execution of the command are displayed in the "failed" sub-window.
33. (Previously presented) The apparatus of Claim 32 wherein the names of the computer systems that have not successfully completed the execution of the command are displayed in red in the "failed" sub-window.
34. (Original) The apparatus of Claim 33 wherein when the displayed name of a computer system is selected further information about the status of the command executing on the computer system is displayed.
35. (Original) The apparatus of Claim 34 wherein if the selected computer system is displayed in the failed sub-window, a reason for the unsuccessful completion of the execution of the command is displayed.
36. (Previously presented) The apparatus of Claim 35 wherein if the selected computer system is displayed in the executing sub-window, a real-time progress of the execution of the command is displayed.

37. (Previously presented) A method of displaying an execution status of a command, the command being executed by a plurality of computer systems on a network, the computer systems running different system management software utilities having different command structures, the method comprising the steps of:

enabling a user to enter the command in a common interface, the command being either a request to start execution of another command or to stop execution of the other command, the common interface translating the command into the different command structures;

enabling a user to send the command to the plurality of the computer systems;

enabling a user to indicate whether or not the execution of the command is to be monitored;

displaying, if the execution of the command is to be monitored, a dialog window that is divided into a waiting, working, successful and failed sub-windows for displaying present status of the execution of the command on each of the computer systems executing the command; and

displaying the status of the execution of the command on each of the computer systems within a proper sub-window.

Appl. No. 09/965,002

Response to Notice of Non-Compliant Appeal Brief dated 01/10/2008

Reply to Office Action of 12/10/2007

(ix)

Evidence Appendix

None.

Appl. No. 09/965,002

Response to Notice of Non-Compliant Appeal Brief dated 01/10/2008

Reply to Office Action of 12/10/2007

(x)

Related Proceedings Appendix

None.